Pacific University
CommonKnowledge

Volume 2 (2002)

Interface: The Journal of Education, Community and Values

5-1-2002

Creating Digital Documents Using Optical Character Recognition

Matt Ernst
Pacific University

Creating Digital Documents Using Optical Character Recognition

Creating Digital Documents Using Optical Character Recognition

Posted on **May 1, 2002** by **Editor**

By Matt Ernst <erns0637@pacificu.edu>
Senior, Computer Science Major at Pacific University

Introduction

Optical Character Recogntion (OCR) is one of the most common and useful applications of machine vision technology. Researchers have experimented with programs designed to recognize images of printed characters since at least the 1960s, but it was in the 1980s that OCR systems expanded in use and significance. Improvements in the power and price of software and hardware since the 1980s have made OCR practical and affordable on standard desktop computers.>

OCR can save many hours of labor when it becomes necessary to convert printed materials into electronic format. There are many different motivations for digitizing documents. Digitizing documents in an unfamiliar language enables their automatic (if clumsy) translation by other software packages . Digitizing paper forms allows anyone on the Web to complete and submit them online, saving time, paper, and postage. The volunteers of Project Gutenberg and other e-text projects use OCR to make literature that has fallen into the public domain available electronically and globally, at no charge.

Transferring any printed material into the electronic world means that it gains all the advantages of documents that originated on a computer in the first place. Texts can be searched for certain words, numbers, or phrases. Excerpts or even entire volumes of material can be sent by e-mail or other electronic means far faster than any postal service can operate. Numbers can be manipulated in spreadsheets. Words can be spellchecked by a variety of tools. Formatting and organization are easily and quickly altered in the world of bits.

From Theory to Practice

The MCEL /BCIS lab recently had need for an OCR application. As part of a project on the Korean War, MCEL had been given a box full of typewritten letters written from an American soldier to his mother during the war. Our challenge was to make these many typewritten pages Web-readable.

My first step was to investigate OCR packages and compare them for price, features, and usability. I had a general understanding that Omnipage Proand Xerox Textbridge were popular and powerful OCR programs. However, Omnipage Pro cost a daunting $499.99 and it turned out that Textbridge had been swallowed by the same company that makes Omnipage. Additionally, searching Usenet via Google Groups indicated that a good number of people found a more obscure product called Finereader to be the best OCR package they'd ever used. I visited the manufacturer's English homepageand downloaded the evaluation version of their Macintosh edition. The evaluation version can be launched 30 times before it expires – plenty to evaluate the software's capabilities. It did extremely well when tested with the Korean War letters. It had, in fact, a 100% recognition rate when I scanned in black and white at 600 dpi. The only errors I found in the scanned output were ones that also existed in the original letters.

Given Finereader's excellent recognition and its relatively low price ($129.99), I didn't even need to evaluate the competing solutions. We could have been scanning and recognizing the same day if we had purchased the software online, but MCEL had no credit card to use and therefore had to opt for physical shipment. It took a few weeks for the package to arrive from Russia, but the software, registration number, and manual all made it here safely. Installation was easy and soon it was time to begin scanning in earnest.

Scanning and recognition were fairly tedious for a few different reasons. To begin with, the letters that we were given had been retyped from the originals. Letters would begin and end in the middle of pages. We had to manually break up the text into individual letters for placement on the Web. Second, although Finereader did a very good job of recognizing and retaining the document layout, it wasn't very good at preserving formatting in HTML or plain text formats. We had to send the scanned documents to Microsoft Word to preserve the placement of text, and hope that we could later export HTML from Word. Anyone who has dealt with Word's HTML export before will realize how painful this may become. Finally, our scanning unit was a plain, older flatbed scanner. We had no document feeder attachment to automate the scanning. Human intervention was necessary to place and scan each sheet of paper that we digitized.

The operator would place a sheet of paper, run the scanning and recognition wizard in Finereader, and send the output to Microsoft Word. In Word, the output would be cleaned up (to join successive pages, for example) and saved in a systematically named file whenever an entire letter had been completed. After each page or each few pages, the operator would delete the scan files from within Finereader to preserve disk space. An 8.5″ by 11″ document occupies quite a bit of storage at 600 DPI. Finereader has better recognition capabilities if documents are scanned in grayscale, but we scanned in black and white (lineart) mode to save time. Sending a 1-bit black and white image from the scanner to the computer is considerably faster than

sending an 8-bit grayscale image, especially at higher resolutions. Finereader had no problems with our black and white documents, although less perfect source material might have demanded that we scan in grayscale.

Low-cost student labor made this project work with minimal additional hardware. For more than a few hundred sheets of source material, however, it would likely make sense to buy a document feeder attachment for the scanner, since the computer can then process successive pages with minimal human oversight. Document feeders will only work with loose pages. If you are digitizing books, journals, pamphlets, or any other bound materials, you will either have to invest in some specialized and expensive pieces of hardware, use a lot of human intervention, or (if the material is expendable) carefully cut away the binding with a razor blade to produce pages suitable for a document feeder.

Problems and Pitfalls

The documents that MCEL needed to digitize were practically perfect for OCR to begin with. They had dark printing on white paper. They were produced in a very common and easy to recognize typewriter typeface. They were clean and perfectly aligned on the page – not at all distorted, skewed, or rotated. Their formatting was simple. Their characters were of reasonable size.

Not all documents are going to enter the digital realm so cleanly and flawlessly. OCR programs in general are much, much less capable than human vision when it comes to distinguishing symbols, and they can be thrown off by a variety of problems. Insufficient contrast will make recognition difficult. Many scanning tools or OCR packages have contrast controls built in so you can assure that text is dark and the background is light. OCR has a particularly hard time with smaller characters, especially subscripts and superscripts. The package has to both recognize the unusually small characters and detect that a vertical shift up or down means that a formatting change has occurred. Recognition of smaller characters may be aided by scanning at a higher resolution and scanning in grayscale instead of black and white. Don't bother to scan at a higher resolution than the optical resolution of your scanner. The software interpolation used at higher resolutions will not reveal new information that the OCR program can use.

Unusual typefaces will likely confuse your OCR program as well. Standard serif and sans-serif fonts should present no problem. Unusually wide, narrow, tall, short, bold, thin, or fancy text may be difficult to recognize. Fortunately, not very many documents are printed in a mixture of Edwardian Script ITC and Wingdings.

A good OCR package will save non-empty portions of a document that it cannot recognize as inline graphics (or at least give you the option to do so). Some manual intervention may be required. This may be particularly needed – and tedious – when dealing with mathematical, scientific, and technical documents. Specialized symbols and diagrams will likely be converted to graphics or misinterpreted altogether without human intervention.

OCR will also be thrown off by a dirty scanner bed or dirty documents. If there are marks on the scanned page – from flyspecks, bits of old crumbled paper, greasy fingerprints, coffee stains, crayon, pencil, or anything else – the recognition program may have a very difficult time. A human may instantly know that passages underlined with a pencil weren't originally that way, but the computer doesn't make such a distinction. It will likely attempt to preserve the underlining in its output, or completely mis-recognize the altered characters.

OCR also has a hard time recognizing characters that are printed too heavily – so that they bleed together – or too lightly, so that the thinner portions of characters are broken. Contrast adjustment may help here too, but in some cases considerable manual intervention (or even pure human transcription) may be necessary.

It will be harder for the program to cope with pages that appear distorted or rotated, either through some artifact of the printing process or from the way they were scanned. Some degree of automatic correction can be performed but, as before, human intervention is necessary in more extreme cases.

Conclusion

Optical Character Recognition systems can save a great deal of time when it comes to recreating printed documents in digital form. However, their success and ease of use is proportional to the initial investment made in the system (both hardware and software), the knowledge of the operator, and the condition of the original documents. Successful transcription of documents will result in highly flexible, easily shared, compact electronic representations suitable for a variety of purposes.

This entry was posted in Uncategorized by **Editor**. Bookmark the **permalink [http://bcis.pacificu.edu/interface/?p=2435]** .

3 THOUGHTS ON "CREATING DIGITAL DOCUMENTS USING OPTICAL CHARACTER RECOGNITION"

africa
on **February 4, 2014 at 10:15 AM** said:

Hello, you employed to write very good articles, but the last many articles were kinda boring… I miss your great articles. Past quite a few articles are just a modest out of track!

nigeria entertainment news

on **February 4, 2014 at 10:27 AM** said:

I have mastered some crucial issues through your internet site post. A single other subject I want to talk about is that there are many games available on a market developed specifically for toddler age children. They include pattern acceptance, colors, household pets, and shapes. These normally focus on familiarization as an alternative to memorization. This keeps little children engaged with no sensing like they're studying. Thanks

nigeria entertainment news
on **February 4, 2014 at 10:35 AM** said:

Sorry for ones huge assessment, but I'm extremely loving the article, and hope this, as well as the excellent review some other individuals have written, will aid you resolve if it is the proper option for you.